UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/015,533	12/11/2001	Andrew J. Fish	10559-628001/P11062	4295

8791 7590 04/07/2005

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
HENRY, MATTHEW ALLAN	

ART UNIT	PAPER NUMBER
2116	

DATE MAILED: 04/07/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/015,533	FISH ET AL.	
	Examiner	Art Unit	
	Matthew A. Henry	2116	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 March 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *14 March 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-3, 5-7 and 17-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Harmer (5,835,760) in view of *Extensible Firmware Interface Specification – Draft for Review.*

Regarding Claim 1, Harmer discloses:

A system comprising (Figure 9, Item 200):

central processor (Column 13, Lines 40-41);

a non-volatile memory coupled with the central processor and storing platform firmware (Column 8, Lines 48-49; Figure 5, Items 108 and 104; Item 104 represents a BIOS, Item 108 represents a system ROM); and

a machine-readable medium (Column 13, Lines 43-45; Figure 9, Item 204) coupled with the central processor, the machine-readable medium to be used in initializing the operating environment for the system upon power up (Column 13, Lines 45-47), the machine-readable medium comprising a first set of instructions (Column 9, Lines 49-54; Figure 6A, Item 128) forming at least a portion of the operating environment (Column 9, Lines 26-29), and a second set of instructions (Column 9, Line 40; Figure 6A, Item 120) defining one or more firmware

extensions to enable the system to access the first set of instructions (Column 11, Lines 34-36; Item 124, a component of the first BIOS portion, enables the second portion by running initialization code of the second portion), wherein the one or more firmware extensions comprise a self-describing media module (Column 9, Lines 16-29).

Harmer does not disclose the system including an extensible firmware interface.

The *Extensible Firmware Interface Specification* teaches:

An extensible firmware interface (EFI) comprising data tables having platform-related information (Page 299, Lines 3-7), a loader for an operating environment (Page 9, Figure 1-1; Page 104, Section 4.4) and boot and runtime service calls available to the operating environment (Page 1, Lines 3-4), wherein the EFI enables extension of platform firmware by loading driver and application images, which when loaded, have access to all EFI defined runtime and boot services (Figure 2-1; Page 13, Lines 1-3)

The motivation for the External Firmware Interface includes the benefit of abstraction, such that code may be written for a variety of "hardware devices without having explicit knowledge of the specifics for each device" (Page 4, Lines 3-5)

Accordingly, it would have been obvious to a person of ordinary skill in the art at the time of invention to incorporate an EFI interface as taught by the *Extensible Firmware Interface Specification* with the system disclosed by Harmer for the benefit of permitting faster and easier development of code for a variety of hardware devices.

Regarding Claim 2, Harmer further discloses:

the machine-readable medium comprises a hard disk platter (Column 13, Lines 47-50).

Regarding Claim 3, Harmer further discloses:

the one or more firmware extensions comprise a file system driver to support a file system format not supported by the platform firmware (Column 9, Lines 49-54; Figure 6B, Item 130).

Regarding Claim 5, Harmer further discloses:

the central processor comprises a central processing unit housed in a single chip (Column 1, Lines 31-35).

Regarding Claim 6, Harmer further discloses:

a volatile memory (Column 13. Line 41, Figure 9, Item 106); and

a motherboard coupling the volatile memory, the non-volatile memory and the machine-readable medium with the central processing unit (Figure 9, Item 200; A motherboard by definition connects the main components of a computer system. Although not explicitly mentioned, it is considered inherent to the operation of the system).

Regarding Claim 7, Harmer further discloses:

A machine-readable medium (Column 13, Lines 43-45; Figure 9, Item 204) comprising:

a first set of instructions defining operations for enabling a machine to access a second set of instructions (Column 11, Lines 34-36; Item 124, a component of the first BIOS portion, enables the second portion by running initialization code of the second portion.) comprising at

least a portion of an operating system stored on the machine-readable medium in a format that is unreadable by the machine before the machine loads the first set of instructions (Column 9, Lines 49-54; Figure 6B, Item 130); and

the second set of instructions (Column 9, Lines 49-54; Figure 6B, Item 128).

Harmer does not disclose incorporating an extensible firmware interface.

The *Extensible Firmware Interface Specification* teaches:

wherein the first set of instructions comprises an extensible firmware interface (EFI) comprising data tables having platform-related information (Page 299, Lines 3-7), and boot and runtime service calls available to the system (Page 1, Lines 3-4), wherein the EFI enables extension of platform firmware by loading driver and application images, which when loaded, have access to all EFI defined runtime and boot services (Figure 2-1; Page 13, Lines 1-3)

The motivation for the External Firmware Interface includes the benefit of abstraction, such that code may be written for a variety of "hardware devices without having explicit knowledge of the specifics for each device" (Page 4, Lines 3-5)

Accordingly, it would have been obvious to a person of ordinary skill in the art at the time of invention to incorporate an EFI interface as taught by the *Extensible Firmware Interface Specification* with the system disclosed by Harmer for the benefit of permitting faster and easier development of code for a variety of hardware devices.

Regarding Claim 8, Harmer further discloses:

the first set of instructions comprise one or more extensions to platform firmware capability (Column 13, Lines 63-66).

Regarding Claim 17, Harmer further discloses:

A data processing system (Figure 9, Item 200) comprising:

means for processing instructions and data (Column 13, Lines 40-41);

non-volatile memory means for storing platform firmware (Column 8, Lines 48-49; Figure 5, Items 108 and 104; Item 104 represents a BIOS, Item 108 represents a system ROM); and

mass storage means (Column 13, Lines 43-45; Figure 9, Item 204) providing means for extending platform firmware capabilities during system booting (Column 9, Lines 26-29) before an operating system loader is loaded and run (Column 11, Lines 50-51; Figure 10, Item 170), wherein means for extending platform firmware capabilities comprises an extensible firmware interface (EFI) comprising data tables having platform-related information (Page 299, Lines 3-7), and boot and runtime service calls available to the operating environment (Page 1, Lines 3-4).

The motivation for the External Firmware Interface includes the benefit of abstraction, such that code may be written for a variety of "hardware devices without having explicit knowledge of the specifics for each device" (Page 4, Lines 3-5)

Accordingly, it would have been obvious to a person of ordinary skill in the art at the time of invention to incorporate an EFI interface as taught by the *Extensible Firmware Interface Specification* with the system disclosed by Harmer for the benefit of permitting faster and easier development of code for a variety of hardware devices.

Regarding Claim 18, Harmer further discloses:

the mass storage means comprises an optical disk (Column 13, Lines 47-50; a compact disk is an optical disk).

Regarding Claim 19, Harmer further discloses:

the means for extending platform firmware capabilities comprise a file system driver to support file system format not supported by the platform firmware (Column 9, Lines 49-54; Figure 6B, Item 130).

3. Claims 4 and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Harmer (5,835,760) and *Extensible Firmware Interface Specification – Draft for Review* in further view of *BIOS Updates*.

Regarding Claim 4, Harmer discloses:

the non-volatile memory (Column 13, Lines 41-42, Figure 9, Item 108).

Harmer does not disclose a non-volatile memory that comprises a random access non-volatile memory.

BIOS Updates teaches:

the non-volatile memory comprises random access non-volatile memory (*BIOS Updates*, Paragraph 2, Lines 5-6).

The motivation for using a random access non-volatile memory, in this case an EEPROM, allows for "a ROM that can be erased and re-written" (*BIOS Updates*, Paragraph 3, Line 3). This will allow for updates to be made to the BIOS without requiring physical replacement of ROM chips.

Accordingly, it would have been obvious to a person of ordinary skill in the art to modify the device disclosed by Harmer to incorporate a non-volatile random access memory as described by *BIOS Updates* for the benefit of providing a circuit housing a BIOS that is more readably modifiable.

Concerning Claim 20, Harmer discloses:

A non-volatile memory (Column 13, Lines 41-42, Figure 9, Item 108).

Harmer does not disclose a non-volatile memory that comprises a random access non-volatile memory.

BIOS Updates teaches:

the non-volatile memory means comprises random access non-volatile memory (*BIOS Updates*, Paragraph 2, Lines 5-6).

The motivation for using a random access non-volatile memory, in this case an EEPROM, allows for "a ROM that can be erased and re-written" (*BIOS Updates*, Paragraph 3, Line 3). This will allow for updates to be made to the BIOS without requiring physical replacement of ROM chips.

Accordingly, it would have been obvious to a person of ordinary skill in the art to modify the device disclosed by Harmer to incorporate a non-volatile random access memory as described by *BIOS Updates* for the benefit of providing a circuit housing a BIOS that is more readably modifiable.

4. Claims 9-11 and 13-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Harmer (5,835,760) and *Extensible Firmware Interface Specification – Draft for Review* in further view of Rakavy (5,978,912).

Regarding Claim 9, Harmer discloses:

the portion of an operating system comprises operating data [that] may include, but is not limited to, system configuration information, data, text, passwords, or any other information that may provide some purpose during the start-up of the system (Column 16, Lines 20-24).

Harmer does not disclose specifically this operating data includes an operating system loader.

Rakavy teaches:

The POST reads a block of data from a predetermined location from the boot device, usually the hard disk or a diskette drive, into memory, and passes control to the program in that data block. This program, known as a bootstrap loader, then loads a larger program into memory. If the larger program is properly loaded into memory the boot program passes control to it. The operating system is then initialized and gains control of the CPU" (Column 2, Lines 27-34).

Rakavy provides this as background for the methodology of the "typical startup procedure of an IBM compatible personal computer" (Column 1, Lines 64-66).

This standard behavior would accordingly suggest that it would be obvious to a person of ordinary skill in the art that, though Harmer does not specifically mention an operating system or bootstrap loader, his invention would follow this standard startup procedure and provide such a program because it serves an important purpose during the start-up of the system.

Regarding Claim 10, Harmer further discloses:

the one or more extensions to platform firmware capability comprise a file system driver to support a file system format used to store at least a portion of the second set of instructions (Column 11, Lines 34-36; the file system described consists of giving the first portion of the expansion BIOS the ability to find and read the second portion of the expansion BIOS).

Regarding Claim 11, Harmer further discloses:

the one or more extensions to platform firmware capability comprise glyphs that represent a language (Column 57-62, glyphs are graphical in nature).

Regarding Claim 13, Harmer discloses:

A machine-implemented method for extending platform firmware capabilities (Column 8, Lines 41-44), the method comprising:

loading on a system one or more firmware extensions (Column 8, Lines 41-44) from a boot media (Column 46-47);

booting the system (Column 13, Lines 53-56); and

loading and running operating data [that] may include, but is not limited to, system configuration information, data, text, passwords, or any other information that may provide some purpose during the start-up of the system from the boot media (Column 16, Lines 20-24) using the one or more loaded firmware extensions (Column 15, Lines 49-53).

Harmer does not disclose specifically the loading and running an operating system loader from the boot media using the one or more loaded firmware extensions.

Rakavy teaches:

The POST reads a block of data from a predetermined location from the boot device, usually the hard disk or a diskette drive, into memory, and passes control to the program in that data block. This program, known as a bootstrap loader, then loads a larger program into memory. If the larger program is properly loaded into memory the boot program passes control to it. The operating system is then initialized and gains control of the CPU" (Column 2, Lines 27-34).

Rakavy provides this as background for the methodology of the "typical startup procedure of an IBM compatible personal computer" (Column 1, Lines 64-66).

This standard behavior would accordingly suggest that it would be obvious to a person of ordinary skill in the art that, though Harmer does not specifically mention an operating system or bootstrap loader, his invention would follow this standard startup procedure and provide such a program because it serves an important purpose during the start-up of the system.

Harmer and Rakavy do not teach firmware extensions being compatible with an extensible firmware interface.

The *Extensible Firmware Interface Specification* teaches:

An extensible firmware interface (EFI) comprising data tables having platform-related information (Page 299, Lines 3-7), a loader for an operating system (Page 9, Figure 1-1; Page 104, Section 4.4) and boot and runtime service calls available to the operating system (Page 1, Lines 3-4), wherein the EFI enables extension of platform firmware by loading driver and

application images, which when loaded, have access to all EFI defined runtime and boot services, the system having an EFI architecture (Figure 2-1; Page 13, Lines 1-3)

The motivation for the External Firmware Interface includes the benefit of abstraction, such that code may be written for a variety of "hardware devices without having explicit knowledge of the specifics for each device" (Page 4, Lines 3-5)

Accordingly, it would have been obvious to a person of ordinary skill in the art at the time of invention to render the firmware extensions to be compatible with an EFI interface as taught by the *Extensible Firmware Interface Specification* with the system disclosed by Harmer for the benefit of permitting faster and easier development of code for a variety of hardware devices.

Regarding Claim 14, Harmer further discloses:

loading one or more firmware extensions from a boot media during a system boot in such a manner that abstracts a mass storage device containing the boot media (Column 13, Lines 47-50).

Harmer does not disclose the method for this abstraction as incorporating a block input/output protocol.

Rakavy further teaches:

POST reads a block of data from a predetermined location from the boot device, usually the hard disk or a diskette drive (Column 2, Lines 27-29).

Regarding Claim 15, Harmer further discloses:

the one or more firmware extensions comprise a file system driver to support a file system format used to store data on the boot media (Column 11, Lines 34-36; the file system described consists of giving the first portion of the expansion BIOS the ability to find and read the second portion of the expansion BIOS).

Regarding Claim 16, Harmer further discloses:

the one or more firmware extensions further comprise glyphs that represent a language (Column 57-62, glyphs are graphical in nature).

5. Claim 12 is rejected under 35 U.S.C. 103(a) as being unpatentable Harmer (5,835,760), *Extensible Firmware Interface Specification – Draft for Review* and Rakavy (5,978,912) in further view of *Unicode Technical Report #10.*

Regarding Claim 12, Harmer discloses:

This general concept of storing information required during the start-up of the system may include a variety of operating data, text, or other information that increases the functionality of the system during the start-up of the system (Column 15, Lines 54-57).

Harmer does not disclose the inclusion of a Unicode collation module as an extension to a system that may be stored on a mass memory storage device.

However, *Unicode Technical Report #10* shows a Unicode Collation Algorithm is a well known method for providing alphabetic, diacritic and case ordering (Page 2, Section "Summary", Paragraph 3, Lines 4-6).

The motivation behind ordering/collation is that sorted entities are far more searchable than ones that are not.

Sorting is a fundamental task in computers and it would be obvious to a person of ordinary skill in the art to modify Harmer to incorporate a Unicode collation algorithm as a method of increasing the functionality of a computer system "without increasing the cost of the peripheral device and/or the system"(Column 15, Lines 52-53).

Conclusion

6. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Extensible Firmware Interface: booting the new generation of Intel Architecture platforms provides further teachings of the extensible firmware interface and appears directed towards encouraging developers to understand and use the interface (Slide 44).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Matthew A. Henry whose telephone number is (571) 272-3845. The examiner can normally be reached on Monday - Friday (8:00 am -5:00 pm).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lynne Browne can be reached on (571) 272-3670. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

LYNNE H. BROWNE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

MAH